



# **INTEGRATION OF EPICS ASYN INTO NON EPICS ENVIRONMENT**

**PRERANA KANKIYA**

**Brookhaven National Laboratory, New York**

**EPICS COLLABORATION MEETING, 2014**

# Outline

- Problem Statement And Proposed Solution
- Background on RHIC Controls
- RHIC and EPICS controls side by side
- Why ASYN is a good place to start
- Building blocks of the solution
- Resources to benefit from
- How to contribute to EPICS
- Summary

# Problem Statement:

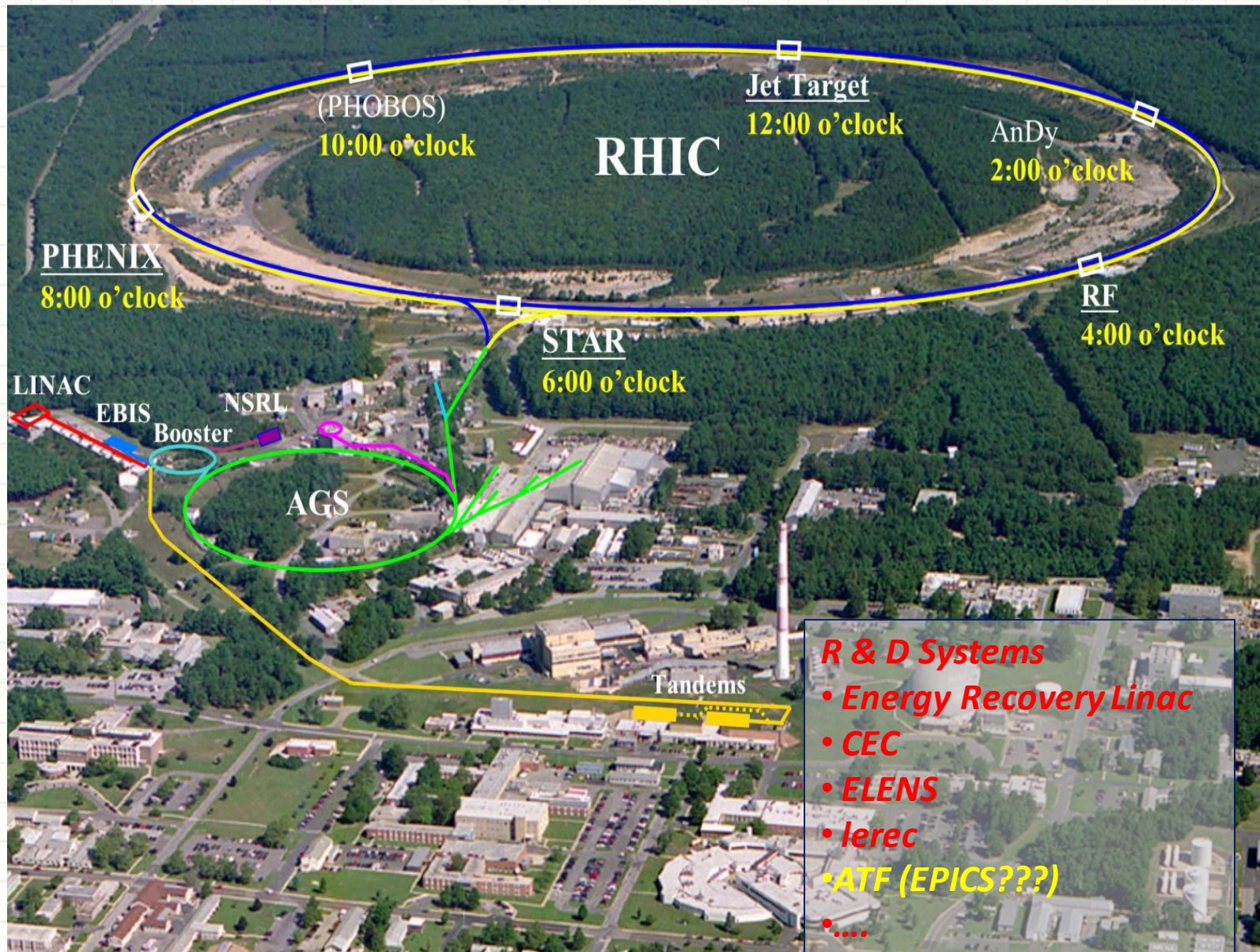
- Variety of hardware interfaces add complexity in control system development. Depth of knowledge required to support each form of communication leads to increase in cost of maintenance and possibility of errors.
- How to increase scope of supported hardware and take advantage of existing resources.



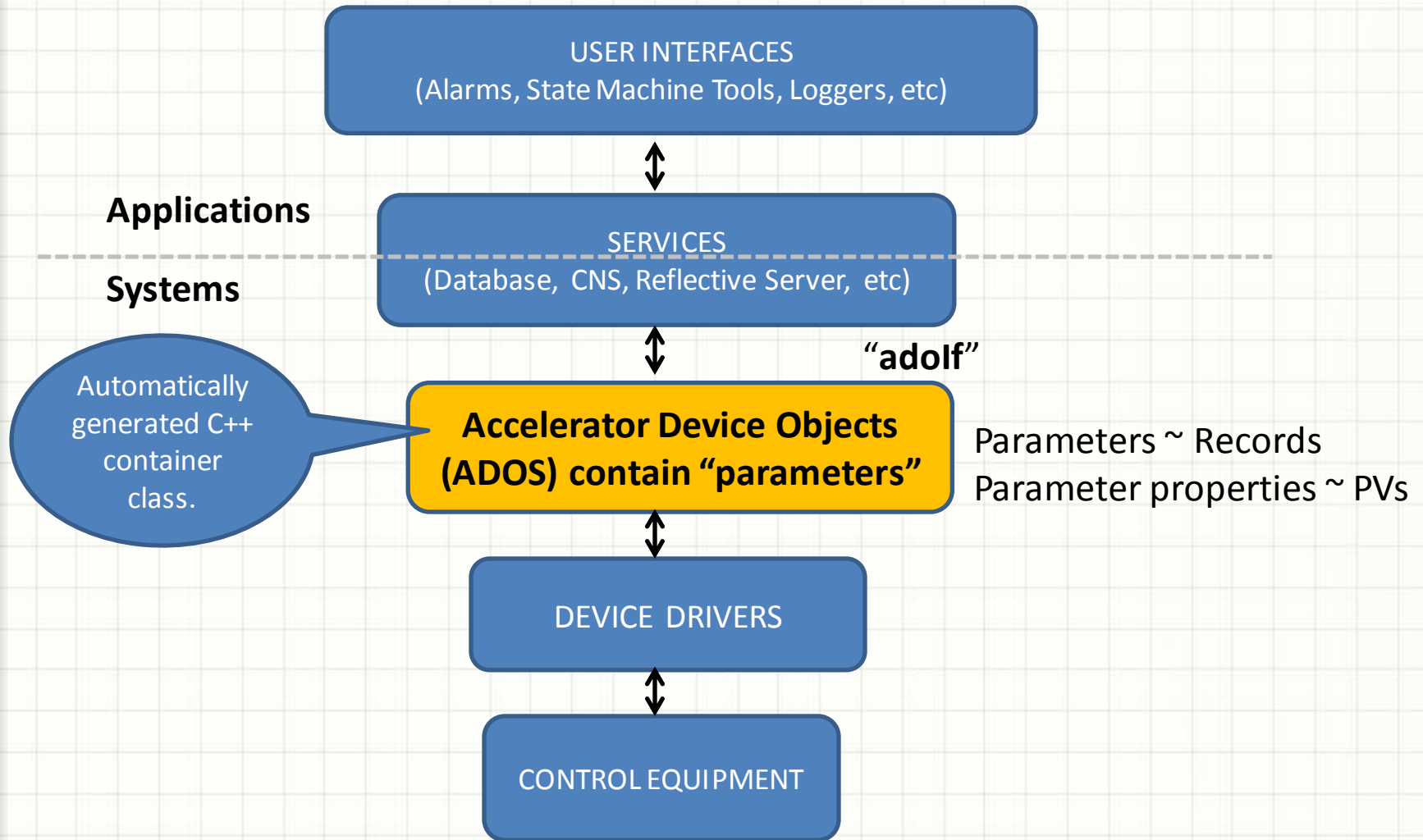
# Proposed Form Of Solution:

- A software layer that abstracts details of difference in protocols and lets higher level code treat all interfaces alike.
- Other Benefits
  - Standardization of code.
  - Off-load some tasks to a common library.
- Possible Approaches:
  - Develop a software library.
  - Adopt existing resources such as ASYN.

# RHIC ACCELERATOR COMPLEX



# RHIC CONTROLS ARCHITECTURE



# RHIC CONTROLS ARCHITECTURE

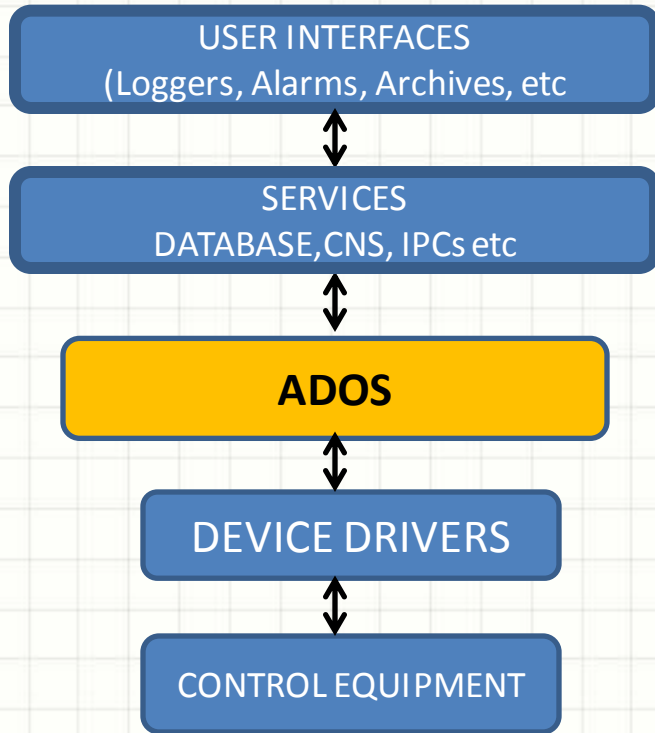
Page PPM Device Data Tools Buffer Help

simple.test	fecName	cscompile01.pbn.
simple.test	description	Examples of para
simple.test	constructTime	1411047893
simple.test	version	1.6 / Sep 17 201
simple.test	className	simple
simple.test	commandBuffer	␣
simple.test	commandBufferLength	0
simple.test	testC	0
simple.test	commandC	commandC
simple.test	menuS	second
simple.test	charS	23
simple.test	ucharS	4
simple.test	shortS	1
simple.test	ushortS	0
simple.test	intS	1
simple.test	uintS	1
simple.test	longS	2000
simple.test	ulongS	1
simple.test	floatS	6e+02
simple.test	doubleS	123

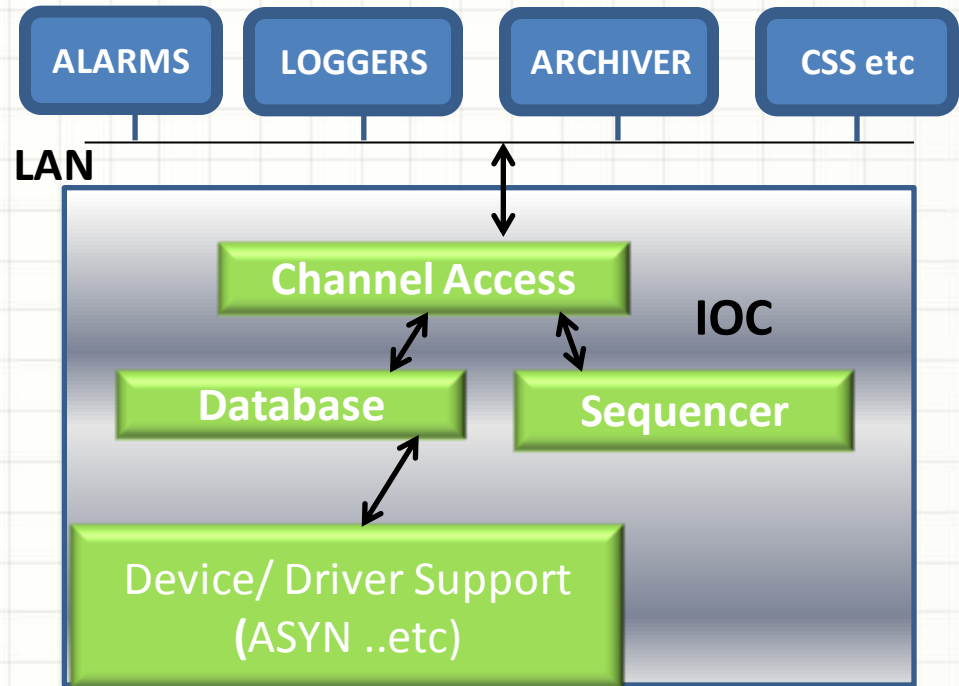
(1,1) ADO Name Nudge: 0 32

Thu Oct 16 12:59:01 2014: copying parameter values to buffer.  
Thu Oct 16 12:59:02 2014: Get and Async requests complete.

# Where to Bridge things with EPICS?



## CONTROL SYSTEM PROGRAMMERS





# Why ASYN is a good place to start

- The two components of ASYN:

- Asyn device support

Integral to EPICS ecosystem and depends on DB definition files which cannot be extracted.

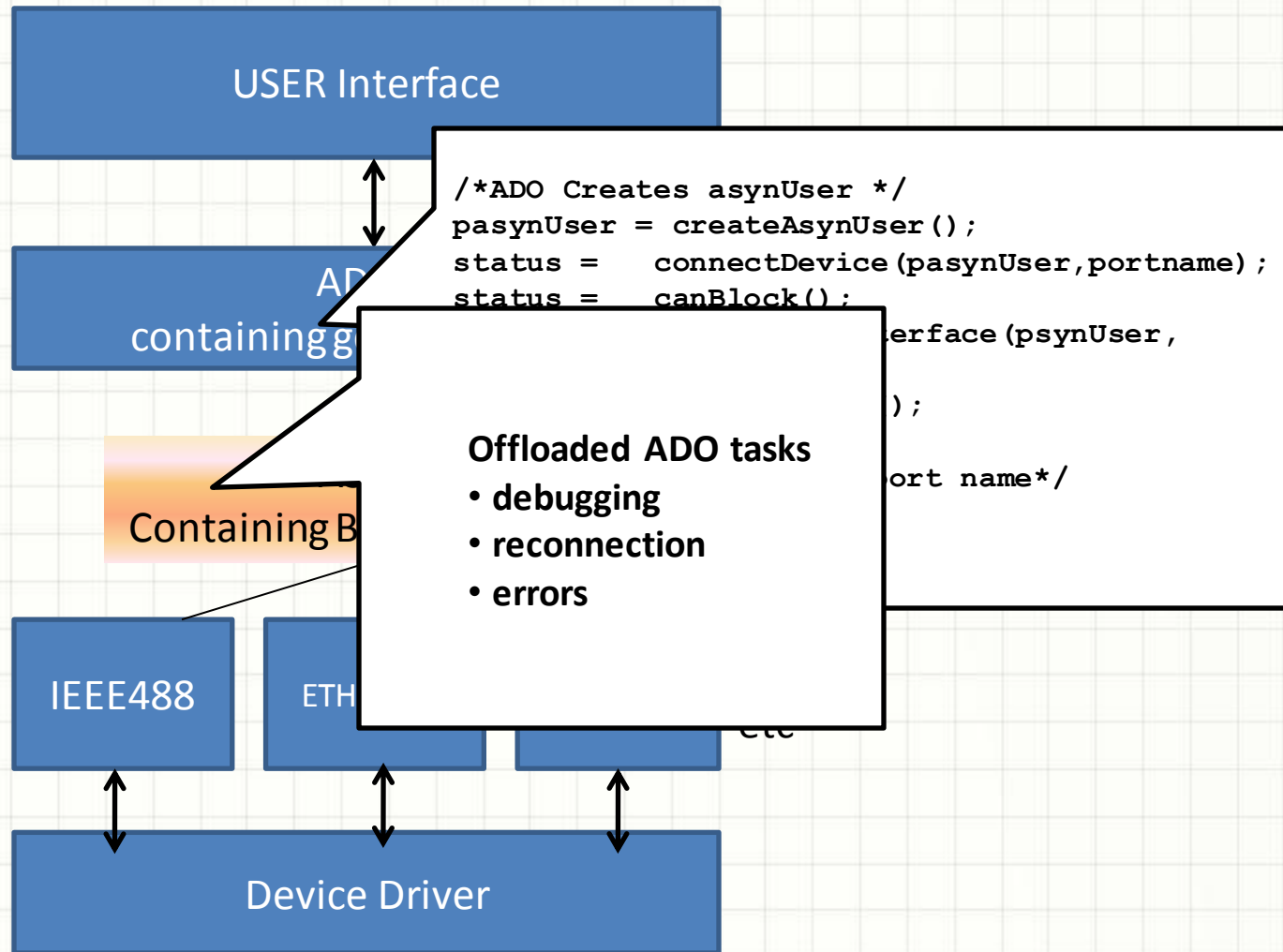
- Asyn drivers

Drivers are completely independent and have regular C style interfaces.

Pros: Asyn supports a large number of devices.

Cons: Not all Asyn supported devices export an Asyn driver.

# Building Blocks of the solution



# Our Drivers of Interest

- Motor Record interfaces from manufacturers such as delta tau and Newport that write drivers for ASYN.
- Area detector package
- Lakeshore 211 Temperature monitors
- Current source/calibrators
- And many more...

# Your Drivers of Interest??

- VME based function generator modules
- COTS power supplies
- SIS3316 a 16 channel VME digitizer of high sampling speeds.
- Vacuum gauge controllers
- a whole lot more....

# Special Use Case

Devices with multiple channels controlled via single IO handle eg Newport esc200

- With help of ASYN's multi user support its easier to create multiple IO handles on the same "port" like resource.
- The ASYN library takes care of locking mechanisms.

# Summary

- There are clearly benefits of EPICS we can take advantage of and there are multiple places we can bridge to EPICS.
- ASYN gives us advantages with minimal EPICS specific dependencies.
- Our driver repository could potentially become relevant to EPICS users.
- ASYN alone does not serve the purpose of a homogenous layer solution.



**THANK YOU !  
QUESTIONS?**